

Manuel Utilisateur (itération 2)

Référence Yummy_MU02 (2002-12-20 - 2003-01-16)
Révision 1.0
Dépendences

Rédaction	Contrôle	Validation
Zoé Drey	Ludovic Courtès	Gérard Dubail
Sébastien Pierre	Ségolène Metais	Vincent Hilaire

Manuel Utilisateur
Itération 2

A propos de ce document

Ce document constitue le manuel utilisateur de la deuxième itération de Yummy. Il est actuellement incomplet car les différents aspects ergonomiques n'ont pas encore été complètement spécifiés, néanmoins la trame du manuel utilisateur est déjà présente pour permettre une mise à jour au fur et à mesure des développements.

Table des matières

Manuel Utilisateur
Itération 2

1. Introduction

Yummy est un environnement de modélisation UML développé en tant que *logiciel libre*. Il vient combler un manque qui existe entre les logiciels de modélisations *lourds* tels que ArgoUML[1] et les logiciels plus généralistes comme Dia[2].

Ce document est la manuel d'utilisation de Yummy, qui présente tout d'abord comment obtenir et installer Yummy. Ensuite une section est dédiée à la prise en main rapide de Yummy (pour ceux qui ne veulent pas lire tout le manuel du premier coup!), puis l'utilisation de chaque type de diagramme est détaillée.

2. Installer et démarrer Yummy

Yummy est librement téléchargeable à partir de son site web accessible à l'adresse `http://projects.type-z.org/yummy`, en cliquant sur le l'icône disquette présente sur la page vous aurez accès à la dernier version de Yummy. Comme tout logiciel libre, vous avez le choix de télécharger soit les binaires (le programme exécutable) et les sources (le programme à compiler).

Il faut savoir avant tout que Yummy est écrit en Java et qu'il nécessite donc soit la machine virtuelle Java seule (JRE) dans sa version 1.3 au minimum, ou bien le kit de développement (JDK) également dans sa version 1.3 au minimum.

2.1. Installation et démarrage des binaires

Une fois l'archive de la dernière version de Yummy téléchargée, décompressez là dans le répertoire où vous voulez que Yummy réside. Vous devriez voir un répertoire nommé *Yummy-X.X* où *X.X* désigne le numéro de version.

Si vous êtes sous Macintosh ou Windows, il vous suffit de double-cliquer le fichier *Yummy.jar* situé dans le dossier de l'archive décompresser. Sous Linux, il suffit d'aller dans ce même dossier et de taper `java -jar Yummy.jar` en ligne de commande, en tenant compte du *y* majuscule.

[1]Que l'on peut obtenir à l'adresse http://argouml.tigris.org

[2]Que l'on peut également obtenir à l'adresse http://dia.sf.net

Vous devriez voir apparaître la fenêtre de démarrage, vous êtes donc en mesure de passer à l'étape suivante.

2.2. Installation des sources et compilation

Une fois l'archive de la dernière version des sources de Yummy téléchargée, décompressez là. Pour compiler Yummy à partir des sources il faut avoir le JDK installé ou bien le compilateur *jikes* d'IBM, mais également l'outil Ant disponible sur `http://jakarta.apache.org/ant`.

Une fois *ant* installé il vous suffit d'ouvrir un terminal et d'aller dans le répertoire où vous avez décompressé Yummy, puis de taper `ant run` pour compiler et lancer Yummy.

Vous êtes maintenant prêt à passer à la section suivante.

3. Prise en main rapide de Yummy

Lorsque Yummy est démarré, vous êtes en présence de la fenêtre représentée par la figure 1.

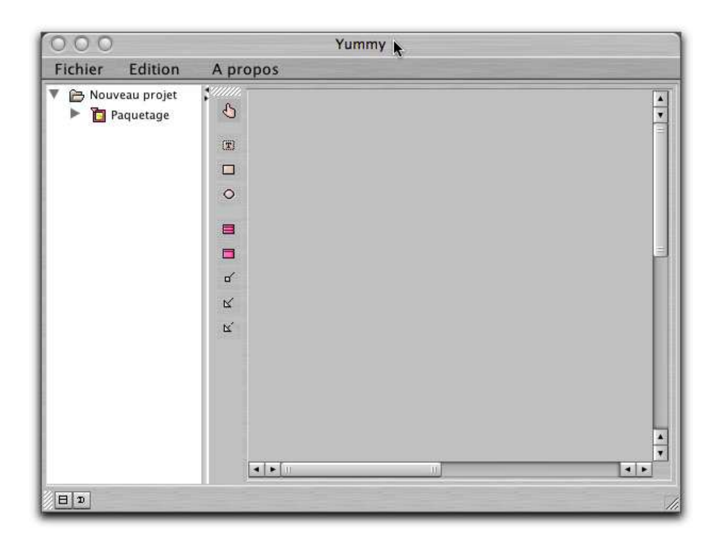


Figure 1. L'éditeur Yummy au démarrage

La fenêtre principale où Yummy est ouvert est la *fenêtre de projet*, cette fenêtre est divisée en deux parties:

1. *Le gestionnaire de projet*
   C'est la partie gauche de la fenêtre, qui représente le projet sous forme arborescente. Nous allons expliquer par la suite la notion de projet.
2. *L'éditeur de diagramme*
   C'est la partie droite de la fenêtre, c'est dans cette partie que seront affichés les diagrammes édites.

On observe par ailleurs la barre de menus, la barre d'édition qui borde verticalement le côté gauche de l'éditeur de diagramme. Nous allons présenter au cour de ce tour guidé de Yummy les différents concepts de Yummy, notamment les notions de projet, diagramme et modèle, qui ont un sens plus prononcé que pour un éditeur de diagramme classique. Nous verrons ensuite comme créer facilement un diagramme de classe et un diagramme de cas d'utilisation, comme sauver et comment charger un projet.

3.1. Les notions de projet, de diagramme et de modèle

Avant de rentrer tout de suite dans la pratique, il est important de préciser quelques notions, dont voici les définitions:

– *Modèle*
  le modèle est la représentation abstraite d'un projet informatique pouvant être modélisée à l'aide du formalisme UML. Le modèle d'un projet est donc l'ensemble des classes, interfaces, acteurs, cas d'utilisation, etc. représentés par différents diagrammes.
– *Diagramme*
  un diagramme est une représentation partielle d'un modèle ; il est composé d'éléments du modèle appartenant à une même famille sémantique au sens d'UML (on parlera ainsi de diagramme de classe, de séquence, de paquetages). Rappelons qu'un diagramme n'est qu'une *représentation du modèle*, il faut donc éviter de confondre les deux, ce qui est une erreur assez courante.
– *Projet*
  Un projet est une collection de modèles et de diagrammes. Les modèles peuvent avoir des origines externes, comme par exemple le modèle de l'API du Java Development Kit, ou bien de modèles internes, tels que les modèles de sous-projets. La gestion de projet au sens de Yummy est donc la gestion de ces différents modèles et des diagrammes qui les représentent.

En résumé, Yummy sert à créer un *modèle UML* qui est représenté par un ensemble de diagrammes. Nous verrons par la suite comme de manière pratique on peut éditer ce modèle au moyen des diagrammes ou du gestionnaire de projet.

3.2. Création d'un premier diagramme de classe

Revenons à notre fenêtre de projet. Un bref coup d'oeil à la barre d'outils nous montre dans l'ordre:

– l'icônes de sélection ( )
– les icônes pour créer du texte ( ), un rectangle ( ), ou une ellipse ( )
– les icônes pour créer une classe ( ), ou une interface ( )
– lec icônes pour créer une agrégation ( ), une généralisation ( ), ou une réalisation ( )

Le projet que Yummy a créer par défaut contient donc également par défaut un diagramme de classe vide que nous sommes en mesure d'éditer é présent.

Avant tout, nous pouvons remarquer qu'un léger espace sépare l'icône de sélection, de icône de dessin de texte et de formes primaires, des icônes de dessin spécifiques au diagramme. En effet, et c'est une constante aux travers des différents diagrammes, nous retrouverons toujours l'organisation suivante de la barre d'outil:

1. En premier lieu les opération de manipulation des éléments du diagramme
2. En second lieu des primitives de dessin
3. En troisième lieu l'ensemble des outils spécifiques au type de diagramme édité.

Trêves de discours, nous allons passer à la réalisation d'un diagramme de classe très simple.

3.2.1. Première étape: la création des classes

Nous allons créer trois classes:

- Une classe nommée `Forme` qui représentera une abstraction de forme géométrique.
- Une classe nommée `Rectangle` et une classe nommée `Ellipse` qui vont représenter deux spécialisations de la classe forme.

Pour ce faire nous allons tout d'abord sélectionner l'icône correspondant aux classes ( ), puis il faut:

- *Pointer* un endroit du diagramme
- puis *cliquer et glisser* pour créer une classe
- *relâcher* le bouton de la souris pour finir le tracé de la classe

Créez trois classes de la sorte, vous devriez être dans la situation montrée à la figure 2

Figure 2. La création de classes

Maintenant cliquez dans le gestionnaire de projet sur l'icône *paquetage* ( ) pour afficher son contenu. Vous devriez voir trois icônes de classe toutes nommées *classe anonyme*, ainsi qu'une icône nommé *Nouveau modèle*, comme montré sur la figure 3.

Comme nous l'avons défini, le gestionnaire de projet affiche le *modèle* sous-jacent aux diagrammes. En effet, lorsque l'on crée une classe dans un diagramme on crée en fait à la fois la classe dans le modèle, et à la fois sa représentation dans le diagramme.

Il s'agit maintenant de renommer nos classes comme convenu. Pour ce faire on peut soit utiliser le gestionnaire de projet ou bien l'éditeur de diagramme, en *double-cliquant sur le nom de la classe*. Une zone de texte apparaît dans laquelle on peut changer le nom. La figure 4 montre les classes une fois renommées.

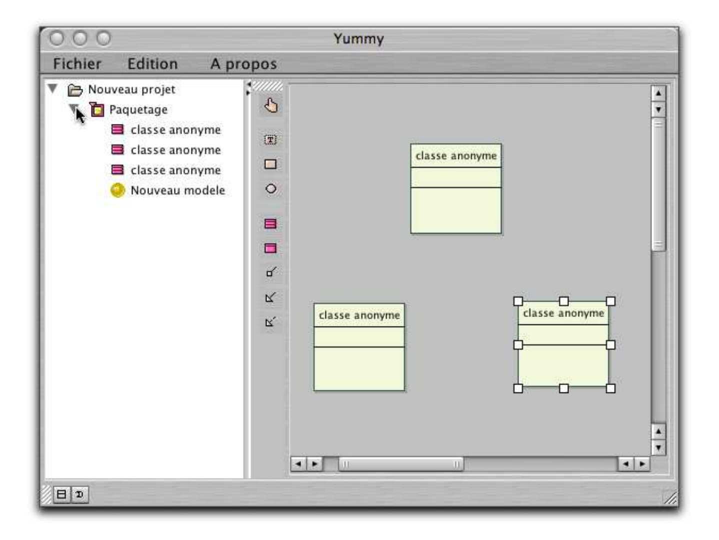


Figure 3. Le contenu du paquetage

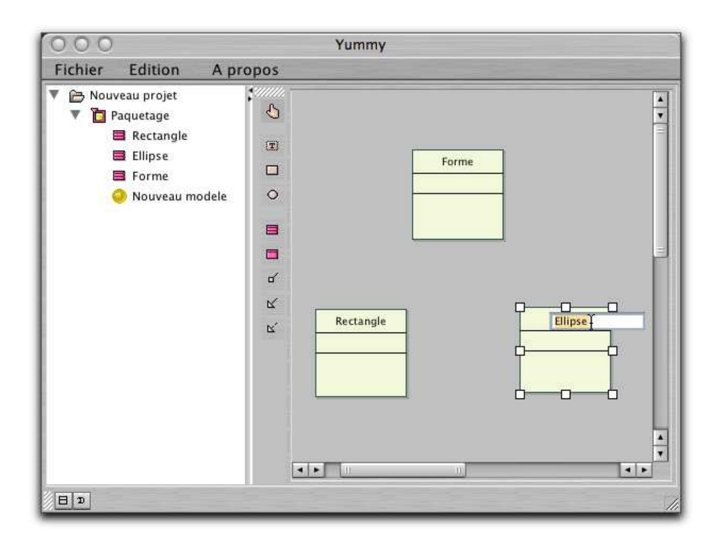


Figure 4. Les classes une fois renommées

3.2.2. Deuxième étape: la création des relations de généralisation

Nous voilà avec nos trois classes, il s'agit maintenant de créer les relations d'héritage (de généralisation pour être précis) entre les classes *Rectangle* et *Ellipse* et la classe *Forme* qui en est bien entendu la classe mère. Pour se faire, il faut tout d'abord cliquer sur l'icône de généralisation ( ), puis:

– Pointer et cliquer sur la classe *Rectangle*
– Glisser le pointeur jusqu'à la classe *Forme*

– Relâcher le bouton de la souris

Durant le déplacement de la souris entre la classe *Rectangle* et la classe *Forme*, le lien de généralisation suit le mouvement de votre curseur. Après avoir fait de même pour la classe *Ellipse* vers la classe *Forme*, vous devriez avoir un résultat similaire à celui représenté sur la figure 5.

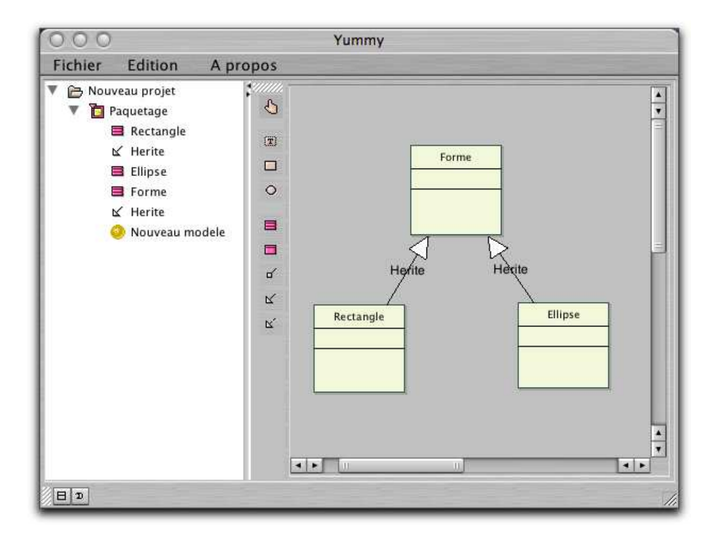


Figure 5. Les généralisations entre les classes

En regardant de plus près le gestionnaire de projet, vous pouvez voir que le modèle a été enrichi de deux généralisations toute deux nommées "Hérite", comme il l'est affiché sur le diagramme.

3.2.3. Troisième étape: sauvegarder et exporter le projet

Maintenant que vous avez réalisé un si beau diagramme, vous voulez peut-être en garder un souvenir, ou même y revenir par la suite (pourquoi pas!). Pour ce faire, Yummy offres différentes possibilités de sauvegardes, comme le montre la figure 6:

– *Enregistrer le projet*
  C'est l'enregistrement le plus complet, puisqu'il enregistre la totalité du projet, modèles et diagrammes. C'est de cette manière qu'il convient d'enregistrer les projets pour une utilisation ultérieure.
– *Exporter le modèle XMI*
  XMI est un format XML pour le modèle UML. Ce format permet d'échanger les modèles faits avec Yummy avec d'autres outils, supportant ce format[1].
– *Exporter le diagramme*
  Permet d'exporter le diagramme, sans exporter le modèle qui va avec. Ceci permet de stocker à part un diagramme que l'on voudrait par la suite réintégrer au projet.
– *Exporter le diagramme en PNG*

[1]Il est à noter que Yummy exporte en XMI 1.1, qui est syntaxiquement différent de la version 1.0, mais bien plus compacte

Permet d'exporter le diagramme au format d'image PNG.

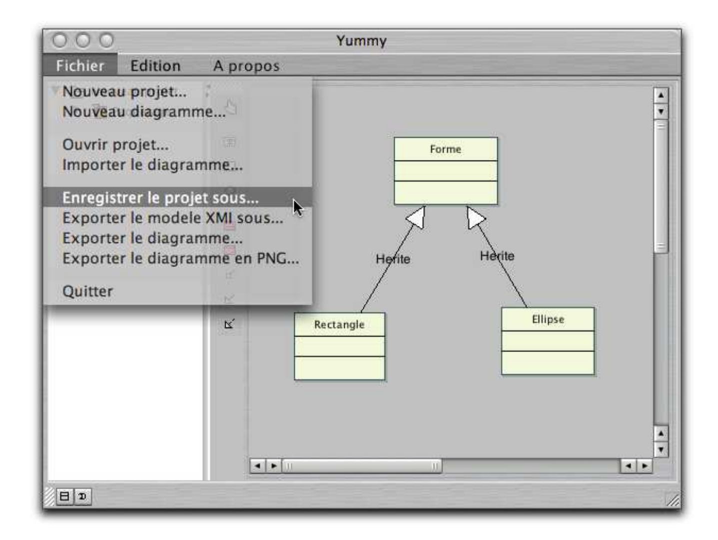


Figure 6. Les menu fichier

Il est donc conseillé de tout d'abord d'enregistrer le projet pour avoir un enregistrement le plus complet, puis d'exporter le diagramme au format d'image PNG.

4. L'édition de diagrammes

Yummy permet d'éditer plusieurs types de diagrammes. Dans sa version actuelle, les diagrammes supportés sont les suivants:

- *Le diagramme de classe*
  Classes, interfaces, généralisation, réalisation, association, agrégation et composition sont gérés.
- *Le diagramme de cas d'utilisation*
  Acteur, cas d'utilisation et dépendance

En plus des éléments purement UML, Yummy permet d'ajouter des éléments graphiques (rectangles, ellipses, textes) qui peuvent venir compléter les diagrammes UML en soulignant des liens non exprimés en UML entre différents éléments.

Nous allons voir dans cette section quelles sont les opérations de base de l'édition de diagrammes, puis nous verrons en détail les différents types de diagrammes offerts.

4.1. Les opérations disponibles dans l'éditeur de diagramme

De manière générale, l'éditeur de diagramme offre les fonctionnalités suivantes:

- *La manipulation des éléments du diagramme*

L'outil sélection ( ) est le seul disponible sur la barre d'outil pour manipuler (déplacer et sélectionner) les éléments du diagramme.

– *Le dessin d'éléments graphiques génériques*
  les outils pour créer du texte ( ), des rectangles ( ), ou encore des ellipses ( ) sont utiles pour compléter un diagramme.
– *La création d'éléments UML*
  Pour créer les éléments UML spécifiques à chaque diagramme, comme les classes ( ), ou encore les acteurs ( ).

Viennent en plus de ces opérations accessibles directement par la barre d'outil, les opérations accessibles par le menu édition:

– *Annuler et rétablir*
  Permet d'annuler ou de rétablir la dernière action effectuée dans l'éditeur de diagramme.
– *Supprimer*
  Supprimer une figure du diagramme. Lorsque l'élément du modèle n'est représenté que dans un seul diagramme, il peut également être retiré du modèle.
– *Grouper et dégrouper*
  Permet de grouper ou de dégrouper des figures de telles sorte qu'il est plus facile d'effectuer des opérations dessus, comme
– *Positionner en dessous ou en dessous*
  Permet de réordonner les figures, de manière à mettre à l'avant plan ou à l'arrière plan le groupe de figures sélectionné.

Nous avons fait le tour des opérations communes à tous les diagrammes. Nous allons donc voir en détail ce que les diagrammes de classes et les diagrammes de cas d'utilisation ont en particulier.

5. Le gestionnaire de projet

Le gestionnaire est sans conteste l'élément le plus difficile à comprendre pour un néophyte. En effet, il représente sous forme arborescente du modèle UML représenté également par les différents diagramme.

Le gestionnaire de projet est utile pour avoir une *vision globale et structurée* de l'ensemble du projet. Le gestionnaire de projet permet de représenter le modèle de différentes manières:

– *Visualisation orientée-modèle*
  Cette visualisation représente le modèle tel qu'il est stocké en mémoire, c'est à dire qu'il n'y a ni réordonnancement, ni introduction de groupes *factices*. C'est la représentation par défaut.
– *Visualisation orientée-diagrammes*
  Cette visualisation représente le modèle de tel sorte que ses éléments sont regroupés dans les diagrammes auxquels ils appartiennent. On pourra notamment voir un même élément apparaître à plusieurs endroits.

Le gestionnaire de projet permet notamment d'effectuer les opérations suivantes:

- *Renommage d'éléments*
  Il est possible de renommer les éléments du modèle, tel que les classes et les associations. Les figures qui les représentent seront automatiquement mis à jour dans les différents diagrammes.
- *Suppression d'éléments*
  Il est possible de supprimer localement un élément d'un diagramme (lorsque l'on est en représentation orientée diagramme), ou bien définitivement du modèle, et donc de tous les diagrammes qui contiennent sa représentation.
- *Partage d'éléments*
  Un élément peut être partagé entre différents diagrammes. Ainsi une même classe pourra être partagée entre différents diagrammes, ceci évitant d'avoir à répercuter les modifications dans chacun des diagrammes.

En résumé, le gestionnaire de projet permet une vision plus claire et structurée du modèle, permettant une édition plus globale qu'avec l'éditeur de diagrammes.

6. Résolution des problèmes

Cette section n'a pas été écrite pour l'instant, mais le sera dans une itération ultérieure.

7. Référence

Cette section n'a pas été écrite pour l'instant, en effet, Yummy étant toujours en cours de développement, le travail ergonomique nécessaire à l'établissement de raccourcis claviers et d'icônes définitives ne sera fait que lors de la 3è itération.

7.1. Outils

La liste des outils disponibles et leur fonction.

7.2. Raccourcis-claviers

La liste des raccourcis clavier par défaut.